SPECTRUM SENSING METHODS

IN COGNITIVE RADIO

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE OF

Bachelor of Technology
In
Electronics and Communication Engineering

Under the guidance of
PROF. SIDDHARTH DESHMUKH

By
SUMIT KUMAR
Roll No. : 111EC0176

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY
ROURKELA, ODISHA
2015

NATIONAL INSTITUTE OF TECHNOLOGY
Dept. of Electronics and Communication Engineering
Rourkela – 769008, Odisha, India

CERTIFICATE

This is to certify that the thesis entitles "Spectrum Sensing Methods in Cognitive Radio" submitted by "Sumit Kumar" (Roll no: 111EC0176) in partial fulfillment of the requirement for the award of Bachelor of Technology Degree in Electronics and Communication at National Institute of Technology, Rourkela is an authentic work carried out by them under my supervision and guidance.

To the best of my knowledge, the matter embodied in thesis has not been submitted to any other university/ institute for the award of any Degree or Diploma.

Place: Rourkela
Date:

Prof. Siddharth Deshmukh
(Supervisor)
Dept. of Electronics and Communication
National Institute of Technology, Rourkela

ACKNOWLEDGEMENT

I wish to express my sincere gratitude towards my project guide Prof. Siddharth Deshmukh, Department of Electronics and Communication, National Institute of Technology Rourkela for providing me with this opportunity to work under him and guiding, supporting and motivating me throughout this project.

I would also like to thank the Department of Electronics and Communication, NIT Rourkela for letting me with everything.

I would also like to thank my friends and seniors who helped me out in clearing some of the conceptual doubts that I had.

SUMIT KUMAR

111EC0176

ABSTRACT

The electromagnetic spectrum is a characteristic asset. The present spectrum authorizing plan is not able to oblige quickly growing demand in wireless communication due to the static spectrum allocation strategies. This allocation prompts increment in spectrum scarcity issue. Cognitive radio (CR) technology is a propelled remote radio design which aims to expand spectrum utilization by distinguishing unused and under-used spectrum in rapidly evolving environments. Spectrum sensing is one of the key strategy for cognitive radio which detects the presence of primary client in authorized licensed frequency band utilizing dynamic spectrum assignment policies to utilize unused spectrum.

In many areas cognitive radio frameworks coexist with other radio frameworks, utilizing the same spectrum yet without creating undue interference. The most simple and easy to implement sensing technique is energy detection. Since, it does not require any prior information of the signal present in the frequency band under observation.

In this thesis, several spectrum sensing technique is introduced but the main work is carried for energy detection methods. Energy detection technique is implemented using VHDL for pseudo random sequence generator.

List of Figures

CONTENTS

Chapter 1

Introduction

1.1 INTRODUCTION TO COGNITIVE RADIO

The radio frequency spectrum is a limited characteristic asset that is divided into spectrum bands. In the course of the most recent century, spectrum bands have been apportioned to diverse services, for example, mobile, fixed, broadcast, fixed satellite, and mobile satellite services. As all the spectrum bands are as of now dispensed to diverse services, most often requiring licenses for operation, a crucial issue confronting future wireless systems is to discover suitable carrier frequencies and bandwidths to take care of the anticipated demand for future services. [1]

With Cognitive Radio being utilized as a part of various applications, the territory of spectrum sensing has become progressively vital. As Cognitive Radio technology is being utilized to provide a method for utilizing the spectrum all the more productively, spectrum sensing is key to this application.

The ability of Cognitive Radio frameworks to get to spare sections of the radio spectrum, and to continue observing the spectrum to guarantee that the Cognitive Radio framework does not create any undue interference depends totally on the spectrum sensing components of the framework.

For the overall framework to work viable and to provide the required change in spectrum efficiency, the Cognitive Radio spectrum sensing framework must have the capacity to adequately recognize some other transmissions, distinguish what they are and inform the central preparing unit inside the Cognitive Radio so that the required actions can be taken.

1.2 MOTIVATION AND OBJECTIVE

CR is an advanced technique which lessens the issue of spectrum scarcity in electromagnetic spectrum. Spectrum sensing is one of the systems which checks the vacancy of primary user designated to particular frequency spectrum. There are a several methods for spectrum sensing for non-cooperative and cooperative CR users. There are few techniques for non-cooperative CR users such as energy detection, matched filter detection, cyclostationary feature detection. Energy detection technique is less complex than matched filter and cyclostationary methods. The energy detection technique does not require any data about the signal structure present in the permitted band to detect the occupancy of user in that band. Energy detection works in high signal – to – noise ratio values compared to other methods.

1.3 THESIS LAYOUT

The thesis is organized in 6 chapters. The current chapter discusses introduction to this in detail, motivation and objective of this work.

Chapter 2 – Introduction to Cognitive Radio

The history, definition, types, characteristics, functions, advantages and disadvantages of cognitive radio are discussed in this chapter.

Chapter 3 - Spectrum Sensing Techniques

In this chapter, different spectrum sensing techniques such as energy detection, matched filter detection, cyclostationary feature detection are discussed. And matched filter detection using MATLAB is discussed.

Chapter 4 – Implementation of Energy Detection Technique using VHDL

Architecture of energy detection is implemented in this chapter. Along with that each components used in the architecture is explained.

Chapter 5 – Results and Discussions

In this chapter, outputs of the energy detection module is discussed. Also ROC curves for random signal in AWGN channel is shown.

Chapter 6 – Conclusions and Future Work

In this chapter brief summary of work done and scope for future work is discussed.

Chapter 2

Introduction to Cognitive Radio

2.1 HISTORY OF COGNITIVE RADIO

The cognitive radio is an emerging technology in wireless communication. It is still too early to tell what a cognitive radio seems to be for different wireless applications due to complexity in implementation of cognitive radio in practical. Therefore, the following history shows the generics of cognitive radio technology.

- In 1998: The concept of cognitive radio was first proposed by Joseph Mitola III in a seminar at KTH (the Royal Institute of Technology in Stockholm).
- In 1999: A comprehensive description of the term cognitive radio was first discussed in a paper written by J. Mitola III and Gerald Q. Maguire.
- In 2000: J. Mitola III wrote his Phd dissertation on cognitive radio as a natural extension of the SDR concept. Mitola described the term cognitive radio as: the point in which wireless personal digital assistants (PDAs) and the related networks are sufficiently computationally intelligent about radio resources and related networks are sufficiently computationally intelligent about radio resources and related computer – to – computer communications to detect user communications needs as a function of use context, and provide resources to radio and wireless services. [2]
- In 2002, the FCC published a report which was aimed at the changes in technology and the profound impact that those changes would have on spectrum policy.
- In 2005, IEEE launched project of 1900 series standard for next generation and spectrum management.
- In 2006, FCC of United States establishes Rule and Order on to use CR devices in unused portions of the TV Whitespaces by secondary basic in 2006.
- The IEEE published 802.22 WRAN (Wireless Regional Area Network) as official standard for CR in 2011.

2.2 COGNITIVE RADIO

Cognitive radio is a type of wireless communication where a transceiver can intelligently distinguish the channels for communication which are being used and which are not being used, and move into unused channels while maintaining a strategic distance from occupied ones. This enhances the utilization of available radio-frequency spectra while interference is minimized to other users. This is an ideal model for wireless communication where transmission or reception parameters of system or node are changed for communication dodging interference with licensed or unlicensed clients. [3]

2.2.1 TYPES OF CR

There are two types of Cognitive Radios:

- Full Cognitive Radio: Full Cognitive Radio (CR) considers all parameters. A wireless node or network can be conscious of every possible parameter observable.
- Spectrum Sensing Cognitive Radio: Detects channels in the radio frequency spectrum. Fundamental requirement in cognitive radio network is spectrum sensing. To enhance the detection probability many signal detection techniques are used in spectrum sensing. [4]

2.2.2 CHARACTERISTICS OF CR

There are two main characteristics of the cognitive radio and can be defined as:

- Cognitive ability: Cognitive Capability characterizes the capacity to catch or sense the data from its radio surroundings of the radio technology. Joseph Mitola initially clarified the cognitive capability in term of the cognitive cycle "a cognitive radio constantly observes nature, orients itself, makes plans, decides, and then acts."

- Reconfigurability: Cognitive capacity offers the spectrum awareness, Reconfigurability regers to radio capability to change the functions, empowers the cognitive radio to be programmed dynamically as per radio environment (frequency, transmission power, modulation scheme, communication protocol).

2.2.3 FUNCTIONS OF CR

There are four major functions of Cognitive Radio. Figure 2.1 shows the basic cognitive cycle:

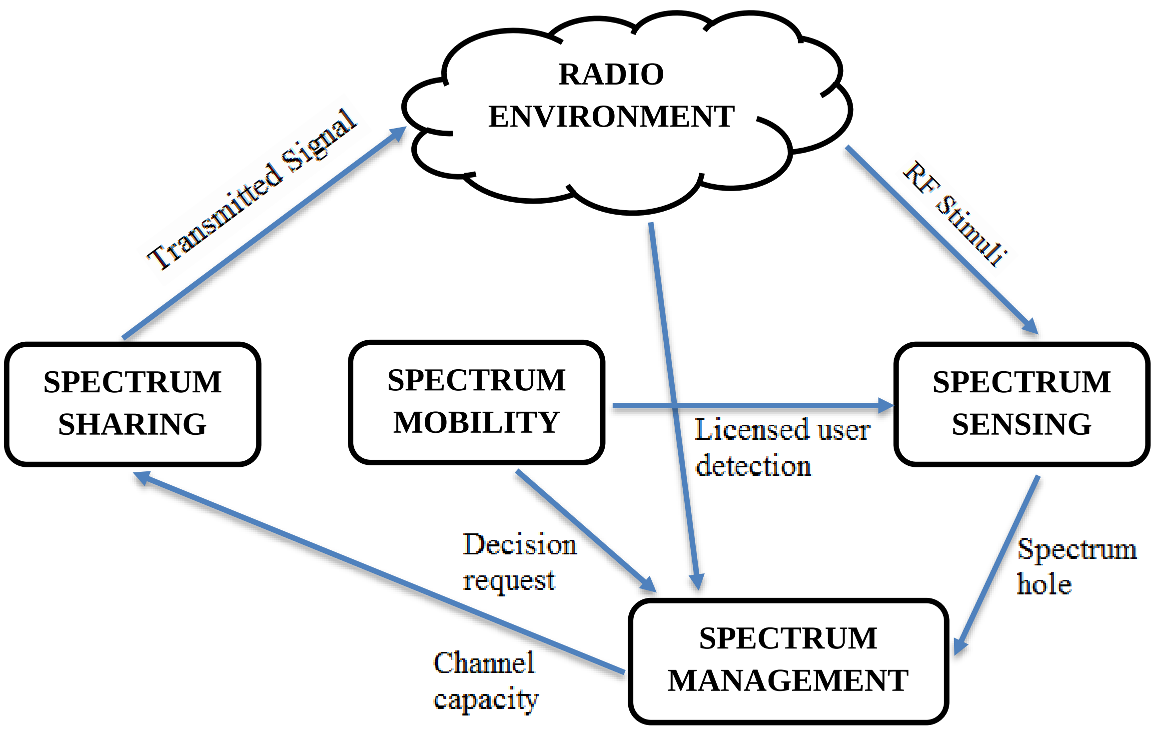

Figure 2.1 Basic cognitive cycle

2.2.3.1 Spectrum Sensing

The principal step of spectrum sensing is that it decides the presence of primary user on a band. The cognitive radio has the capacity to impart the result of its detection with other cognitive radios in the wake of sensing the spectrum. The main objective of spectrum sensing is to discover the spectrum status and activity by periodically sensing the target frequency band.

2.2.3.2 Spectrum Management

Provides the reasonable spectrum scheduling technique among coexisting users. The available white space or channel is quickly chosen by cognitive radio if once found. This property of cognitive radio is described as spectrum management.

2.2.3.3 Spectrum Sharing

Cognitive Radio doles out the unused (spectrum hole) to the secondary user (SU) as long as primary user (PU) does not utilize it. This property of cognitive radio is described as spectrum sharing.

2.2.3.4 Spectrum Mobility

When an authorized (Primary) user is detected, the Cognitive Radio (CR) empties the channel. This property of cognitive radio is depicted as the spectrum mobility, also called handoff.

2.3 APPLICATIONS, ADVANTAGES AND DISADVANTAGES OF CR

2.3.1 Application:

- Improving reliability in wireless communication system.
- Less expensive radio.
- Advanced network topologies.
- Enhancing SDR techniques.
- Automatic radio resources management.

2.3.2 Advantages:

- Mitigate and solving spectrum access issues.
- Spectrum utilization improves.
- Improves wireless networks performance through increased user throughput and system reliability.
- More adaptability and less co-ordination.

2.3.3 Disadvantages:

- Software reliability
- Loss of control
- Regulatory concerns
- Fear of undesirable adaptations.
- Significant research is to be done to commercially use cognitive radio.

Chapter 3

Spectrum Sensing Technique

3.1 SPECTRUM SENSING

The cognitive radio system examines all level of flexibility (time, frequency and space) to predict spectrum usage. There are a few procedures available for spectrum sensing. Spectrum sensing is a system which figures out if a given frequency band is utilized. A wide range of routines are proposed to recognize the presence of signal transmission and can be utilized to improve the detection probability.

3.1.1 Energy Detection

The aim of the spectrum sensing is to decide between two hypotheses which are

$x(t) = w(t), H_0$ (Primary User absent)

$x(t) = h * n(t) + w(t), H_1$ (Primary User present)

Where x(t) is the signal received by the CR user, n(t) is the transmitted signal of the primary user , w(t) is the AWGN band, h is the amplitude gain of the channel. H0 is a null hypothesis, which states that there is no licensed user signal.

Energy Detection is a simple detection method. The energy detection is said to be a blind signal detector in light of the fact that it overlooks the structure of the signal. Energy detection is based on the rule that, at the receiving end, the energy of the signal to be detected is computed. It estimates the presence of a signal by comparing the energy received and a known threshold λ derived from the statistics of the noise.

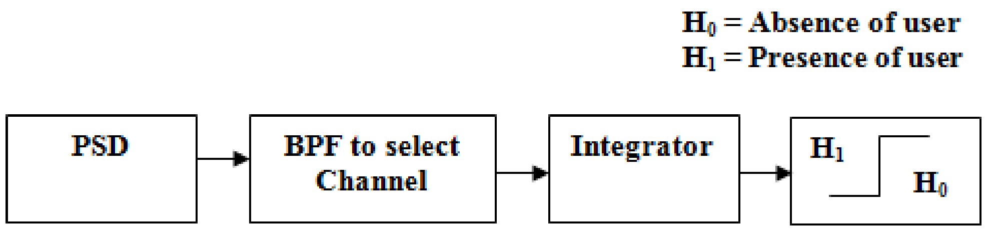

Figure 3.1 Energy Detector Block Diagram

3.1.2 Matched Filter Detection

The best sensing technique in AWGN environment without ant prior information about the signal is ED technique. If we considered the signal structure, then we can get best performance by using matched filter method.

Matched filter is a linear filter which used to maximize signal to noise ratio in presence of additive noise. It provides coherent detection. A coherent detector uses the knowledge of the phase of the carrier wave to demodulate the signal.

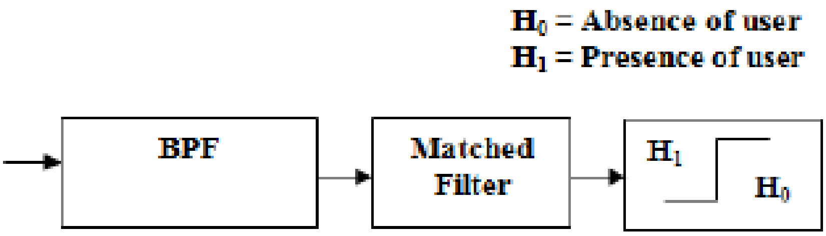

Figure 3.2 Matched filter Block diagram

3.1.3 Cyclostationary Feature Detection

Cyclostationary feature detection taking into account introduction of periodic redundancy into a signal by sampling and modulation. The periodicity in the received primary signal to recognize the presence of Primary Users (PU) is misused by Cyclostationary feature detector which measures property of a signal specifically Spectral Correlation Function (SCF) given by$S_x^{\infty}\ (f)\ = \int_{-\infty}^{\infty} R_x^{\infty}\ (\tau)\ e^{-j2\pi ft}\ d\tau$

Where $R_x^{\infty}\ (\tau)$ is cyclic autocorrelation function (CAF).

Cyclostationary feature detector can differentiate the modulated signal from the additive noise, recognize Primary User signal from noise. It is used at low SNR detection by using the data information embedded in the Primary User signal which does not exist in the noise. This method is robust to noise discrimination and it performs better than energy detector.

It has disadvantage of more computational complexity and more time observation.

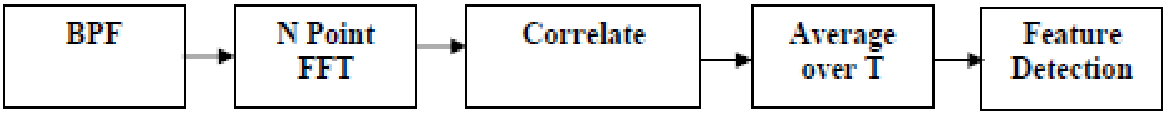

Figure 3.3 Cyclostationary feature detection block diagram

3.2 COGNITIVE RADIO SYSTEM IMPLEMENTATION USING MATLAB

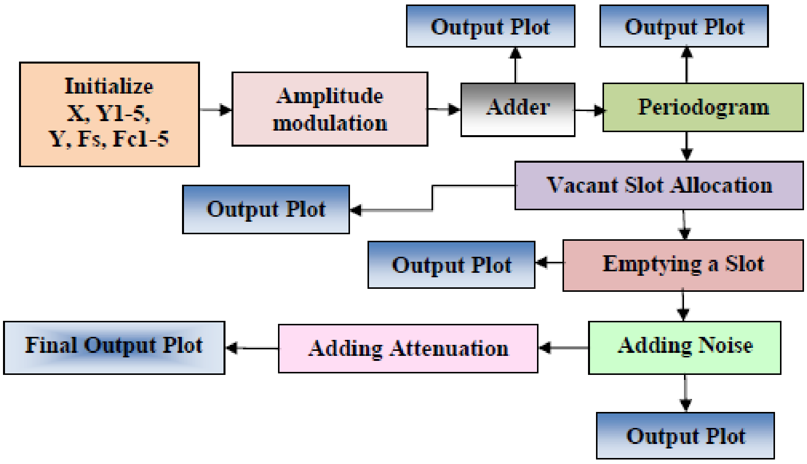

Figure 3.4 Methodology/Block diagram of set up

- **Initialization-** 5 Carrier Frequency Bands for Users, Message Frequency and the Sampling Frequency are initialized.
- **Modulation-** Modulates user data over the respective frequency band by amplitude modulation.
- **Adder-** Addition of all the modulated signals to produce a transmitting signal.
- **Periodogram-** To estimate the power spectral density of received signal.
- **Vacant Slot Allocation-** New User is allotted to the first spectral hole when he arrives.
- **Emptying a slot-** Asked user to empty a specific slot if all the slots are engaged.
- **Addition of noise-** Amount of Noise to be added.

3.2.1 RESULT

The cognitive radio framework ceaselessly looks for the spectrum hole where primary user is not present and is determined by the technique for energy detection. When it figures out the spectrum hole, promptly it allots to the Secondary User (SU) and at whatever point Primary User (PU) wants to possess the slot, Secondary User instantly leaves it.

Note: Code for this result is given in Appendix I

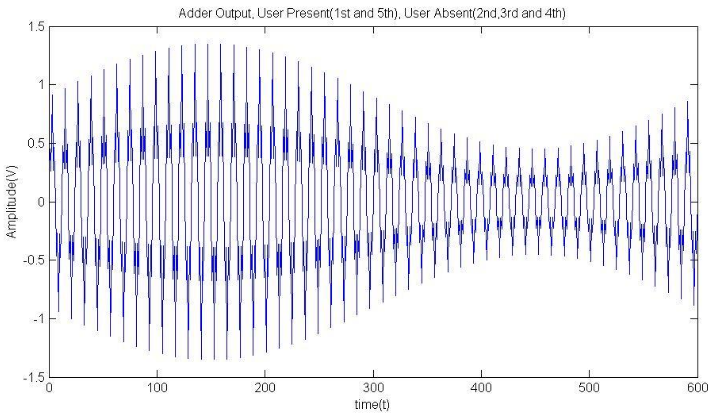

Figure 3.5 Adder Output, User Present (1^{st} and 3^{rd})
User absent (2^{nd}, 3^{rd} and 4^{th})

It has been assumed that 1^{st}, 5^{th} primary users are present and 2^{nd}, 3^{rd} and 4^{th} primary users are not present.

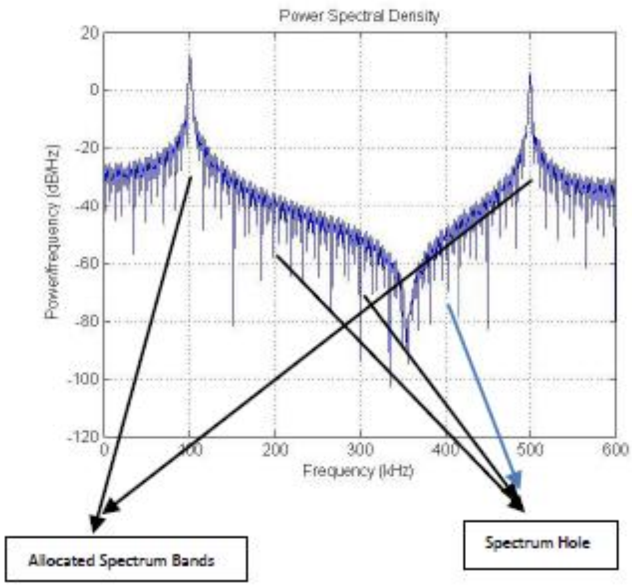

Figure 3.6 Used bands (1^{st} and 5^{th}), Unused bands (2^{nd}, 3^{rd}, 4^{th})

Now the Cognitive Radio (CR) framework will search for the most readily available gap (Spectrum hole) and naturally allot it to the secondary user (SU) in the spectrum. The first available spectral gap was occupied by the secondary user (SU) 1.

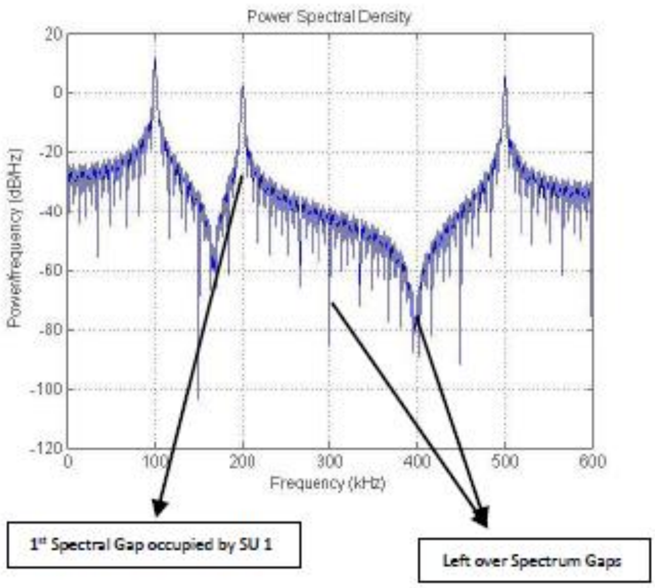

Figure 3.7 1st unused band assigned to SU 1

Now the system will search the next spectrum hole and automatically assign it to the Secondary user (SU) in the spectrum. As shown in above figure, the next available gap was occupied by the secondary user (SU) 2.

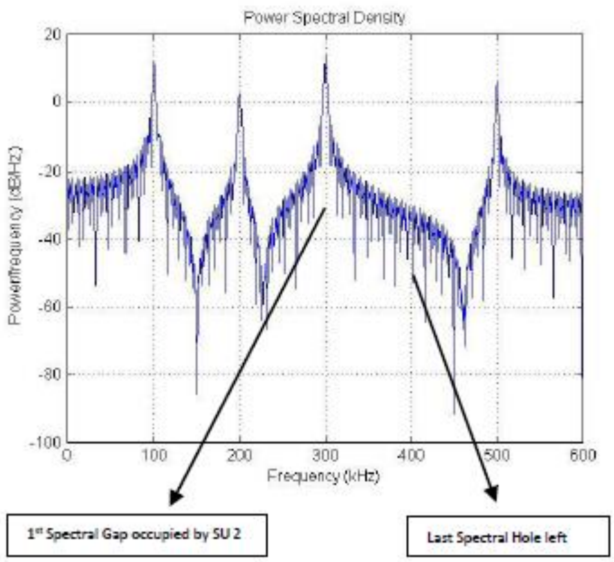

Figure 3.8 2nd unused band assigned to SU 2

Now just one slot left empty which will get filled by addition of another Secondary User (SU) as shown in Figure. Here the majority of the frequency bands are being used productively after the last spectrum gap is filled by secondary user 3.

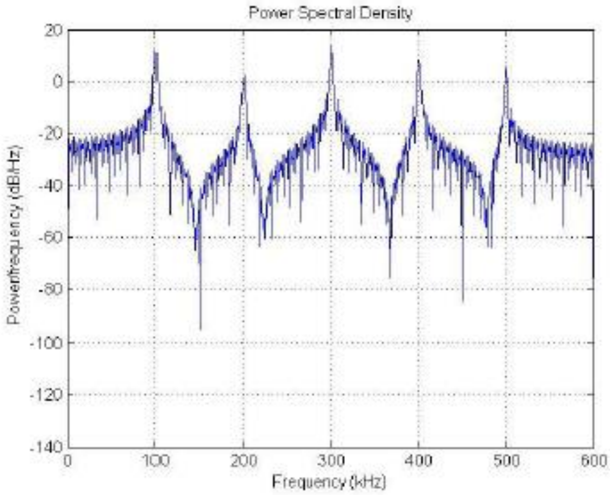

Figure 3.9 All of the spectrum bands are in use

When all the slots are being allocated, system won't enthral other user and will have the capacity to free up the spectral hole (slot) one by one. If asked to empty a slot, it will erase the information in the first spectral gap and make it prepared for the following task.

Chapter 4

Implementation of Energy Detection Technique using VHDL

4.1 INTRODUCTION

In previous chapters we discussed about CR, spectrum sensing techniques and in chapter 3 one of the detection technique i.e. Matched filter detection was implemented using MATLAB and simulations were shown. In this chapter we will discuss about Energy detection and how we can implement it using VHDL. This chapter designed Energy detector module for random generated signals, in this case PRSG.

4.2 ARCHITECTURE OF ENERGY DETECTION TECHNIQUE

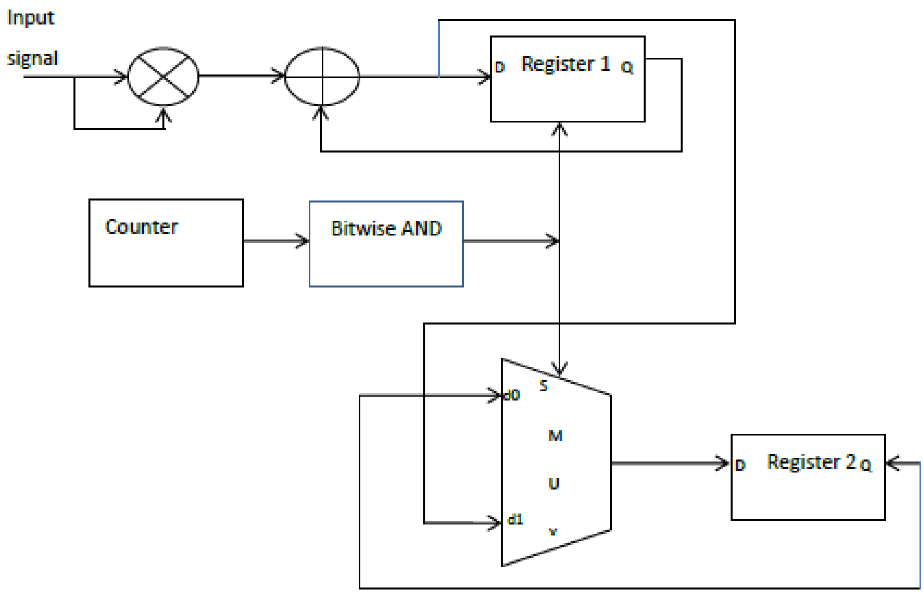

Figure 4.1 Architecture of Energy Detector

In figure 4.1, the architecture of Energy Detector is proposed. As we can see from the figure, it consists of multiplier, adder, register, counter and multiplexer. The input signal is taken to be PRSG (Pseudo Random Sequence Generator).

The inputs samples are passed through the multiplier so as to get the square of the samples. The squared samples are then added and accumulated using an adder–register configuration, as indicated in Figure 4-1. Here distinctive number of samples is taken to get detected energy value. However, the value N can be decided to be higher for better precision. The counters tally from 0 to total number of samples ceaselessly. For Example if the number of samples are taken 16 then, then counter counts and When 16 samples are reached, the contents of the counter turns into 15; then, output of the counter turns into "1," and Register1 is cleared or reset. The select Signal for multiplexer Mux now turns into "1," and the energy value for 16 samples is send to Register2, which forms the output. While counting from 0 to 15, i.e., amid the accumulation of 16 samples, the output of counter is "0;" consequently, the Mux will choose the output of Register2 itself, and henceforth, the output remains consistent. As a consequence of this, independent of the time duration of the input signal, the energy for 16 examples will be obtained at the output; consequently, the proposed energy detector can be utilized for real time applications.

4.2.1 Pseudo Random Sequence Generator

Pseudo Random number generator is a shift register whose input is taken as a random value or a random seed value which then maps it to a longer pseudorandom string. The random seed is typically a short binary string. The only single bit is function of XOR logic. Here first input bit is provided by taking linear XOR function of third and last bit. The other bits are depended on previous bit. The operation then follows to generate random bits. The register has finite number of states and random sequence is repeated after some cycles.

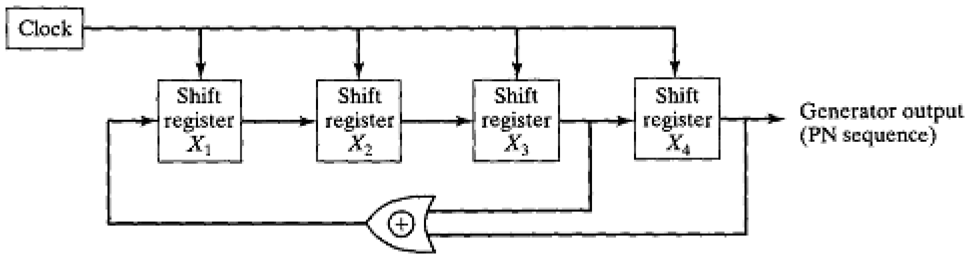

Figure 4.2 XOR operation of PRSG

The sequence generated by PRSG is not theoretically random, but for most practical applications the sequence can be considered as random. The period of the sequence is $2^n - 1$, where n is the number of shift register used. For a 4-bit shift register we obtain the following table, where we see that after 15 samples the values are repeated.

PRSG Stages				Value (3:0)
3	2	1	0	
1	0	0	0	8
0	0	0	1	1
0	0	1	0	2
0	1	0	0	4
1	0	0	1	9
0	0	1	1	3
0	1	1	0	6
1	1	0	1	13
1	0	1	0	10
0	1	0	1	5
1	0	1	1	11
0	1	1	1	7
1	1	1	1	15
1	1	1	0	14
1	1	0	0	12
1	0	0	0	8

Table 4.1 A 4 – bit PRSG

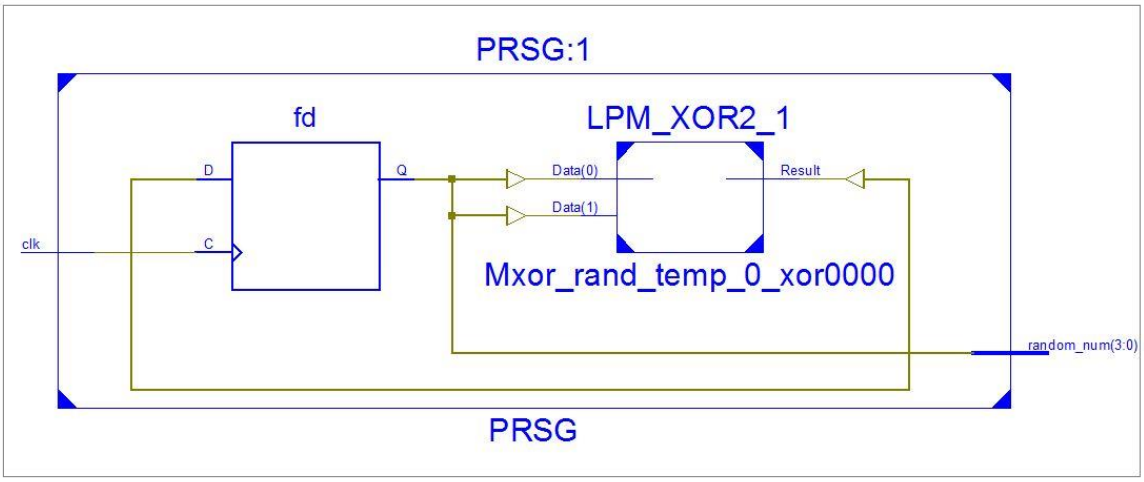

Figure 4.3 RTL schematic of PRSG

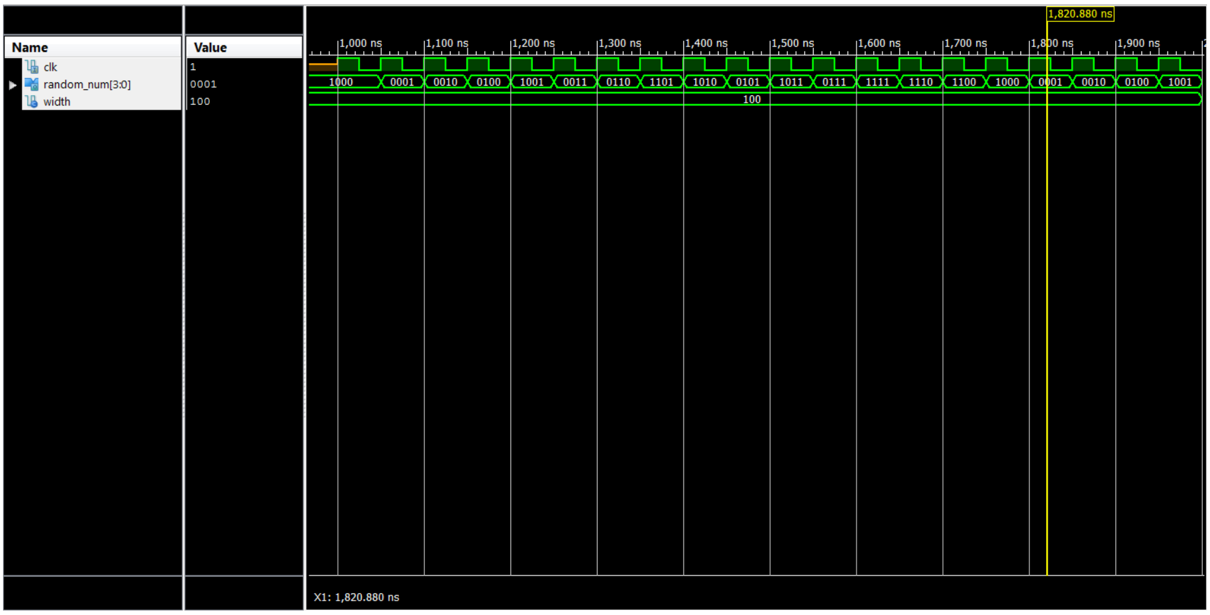

Figure 4.4 PRSG output

4.2.2 Multiplier

Multiplier is an electronic circuit used in digital electronics to multiply two binary numbers. Here, multiplication of two 4-bit number has been performed. The output is an 8-bit binary number.

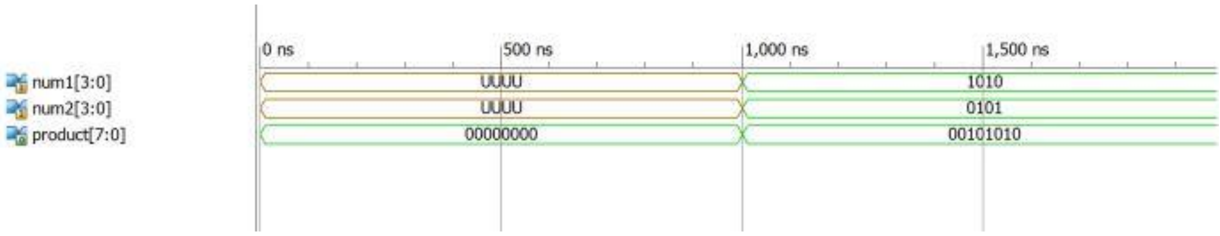

Figure 4.5 Multiplier output

4.2.3 Adder

8-bit adder is used here which adds two data from the multiplier output and the output from the Register 1.

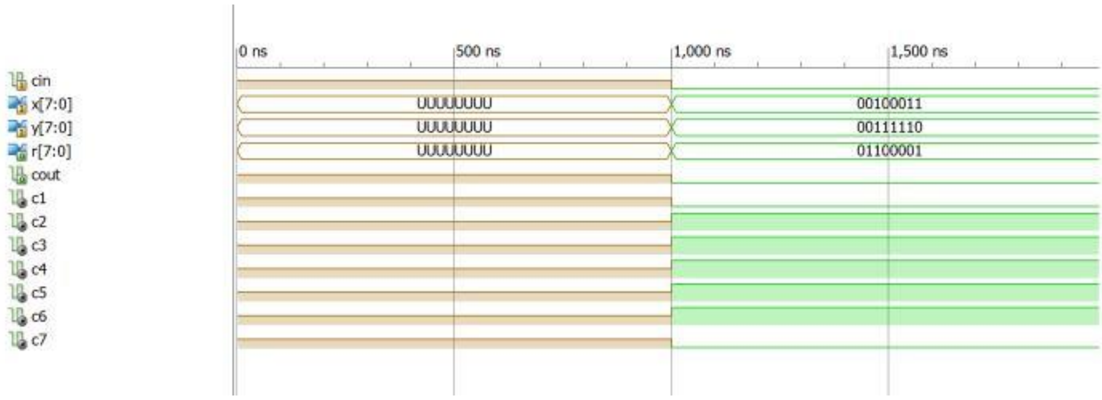

Figure 4.6 Adder output

4.2.4 Register

8-bit register is used. It is basically a D-flip flop register. When ‘rst’ signal is 0, it keeps on accumulating the values which is also reflected at its output. When ‘rst’ signal becomes ‘1’ the values in the register is cleared.

4.2.5 Counter

4 – bit counter which counts from 0 to 15 is used. When the counter value reaches 15, it outputs a logic 0 which is done through programming. This signal is then given to ‘rst’ signal of Register 1 and ‘sel’ signal of MUX.

4.2.6 Multiplexer

A 2:1 multiplexer is used here. When ‘sel’ signal is ‘0’, the mux output is ‘d0’ which is the output from the Register 2, i.e. it shows the value that is stored in the Register 2 from the previous cycle. When ‘sel’ signal is ‘1’, the mux output is ‘d1’ which is the accumulated value of the 15 samples from Register 1. ‘sel’ signal becomes 1 when the counter value reaches 15 which is the threshold value that was initialized in the counter.

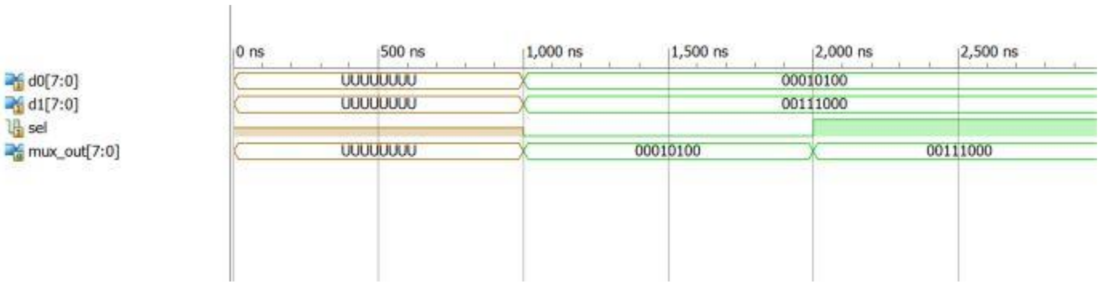

Figure 4.7 Multiplexer output

Chapter 5

Results and Discussions

5.1 RESULTS AND DISCUSSIONS

The PRSG behaviour is checked using ISE simulator. Xilinx 14.2 is used to write VHDL code for detected energy value for PRSG. The input of the energy detector module is ‘clk’, ‘rst’ and output is energy output.

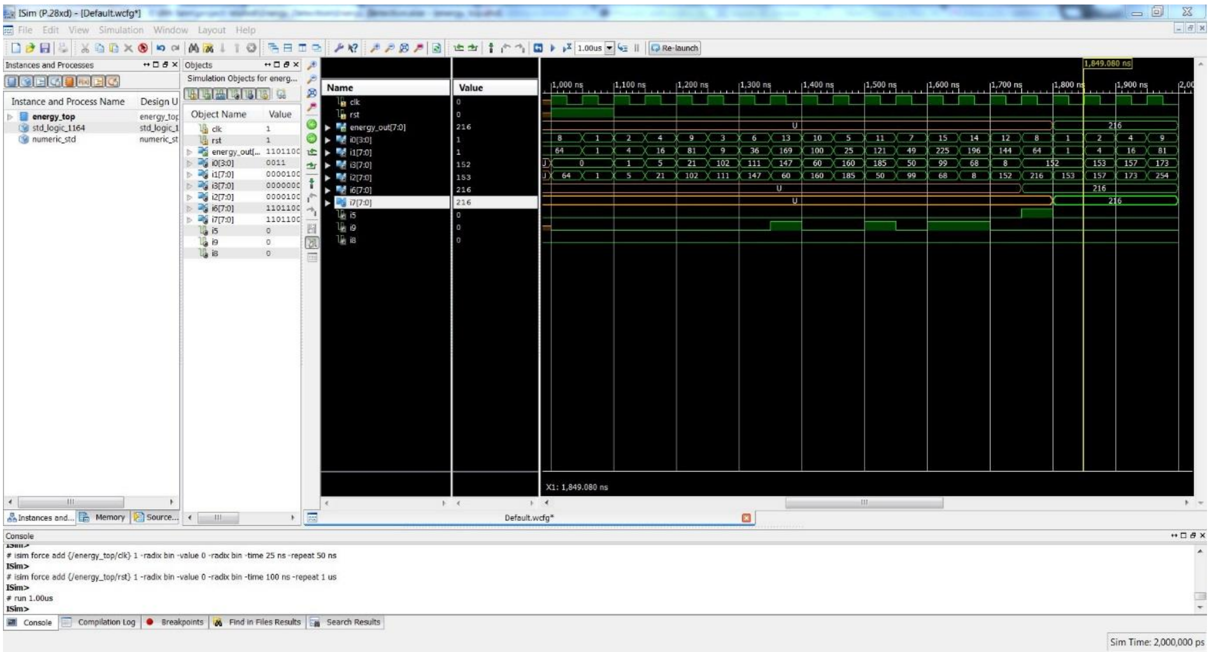

Figure 5.1 Energy Detected value of 16 samples

In this work all the inputs and outputs are taken in form of digital. The PRSG is utilized to create arbitrary 4-bit binary sequence. Figure 5.1 shows the binary sequence generated, its squared output and the detected energy output. The detected energy value is found out using energy detector architecture, where input sequence is produced from PRSG.

Figure 5.2 RTL schematic of top module

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	21	4656	0%
Number of Slice Flip Flops	24	9312	0%
Number of 4 input LUTs	27	9312	0%
Number of bonded IOBs	10	232	4%
Number of MULT18X18SIOs	1	20	5%
Number of GCLKs	1	24	4%

Table 5.1 Design summary of energy detection module

5.2 ROC CURVES

ROC curves for random signal in AWGN channel is calculated using MATLAB. The results obtained are shown in following figures.

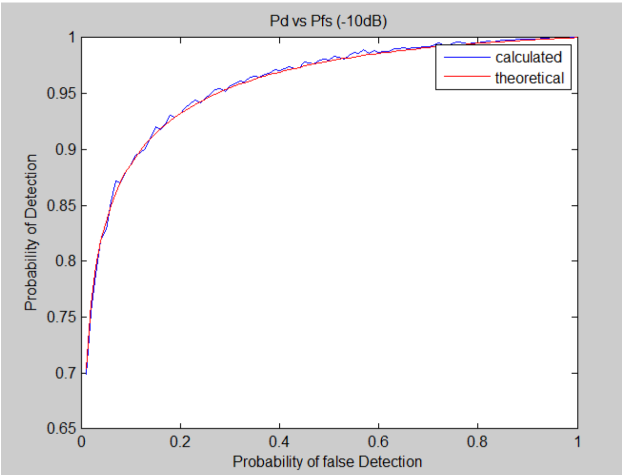

Figure 5.3 Pd vs Pf (at -10 dB)

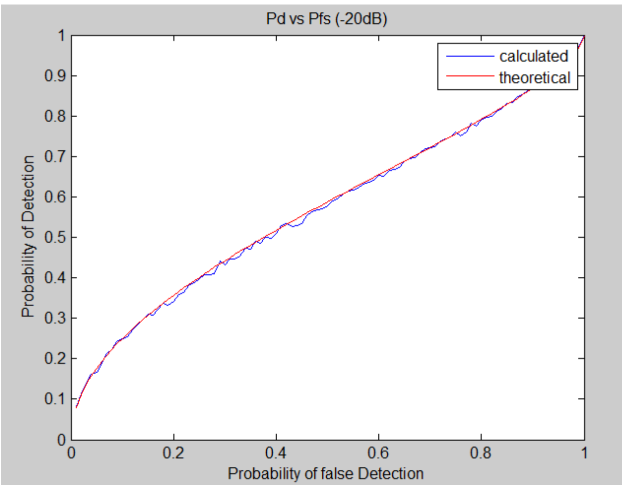

Figure 5.4 Pd vs Pf (at -20 dB)

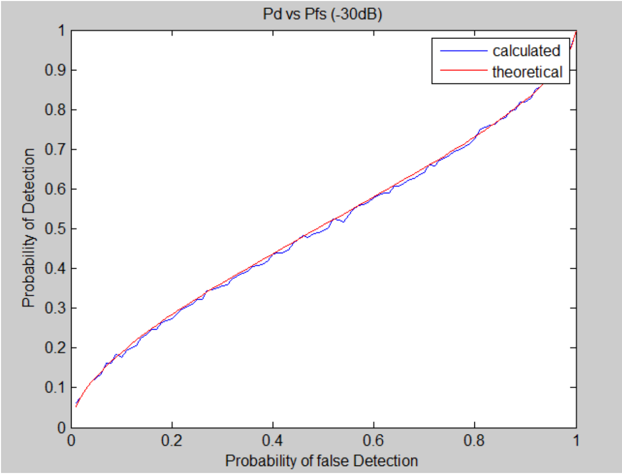

Figure 5.5 Pd vs Pf (at -30 dB)

The ROC curves shown is for different SNR values. It is seen that probability of detection is different and detection quality is good at higher SNR values. The code for finding out the curves is given in Appendix.

Chapter 6

Conclusion and future work

6.1 CONCLUSION

This thesis gives some idea regarding cognitive radio technology, its diverse classifications and distinctive spectrum sensing methods. The work of this thesis contributes toward energy detection procedure and finally method executed using VHDL code.

All the works in this thesis are based on MATLAB and VHDL simulation. The simulation results are taken for different number of samples to study energy detection performances.

6.2 FUTURE WORK

This thesis has been completed in the final year of B. Tech and hence the hardware implementation could not be carried out in the tenure. Thus it can be carried out in future.

REFERENCES

[1] M. Matinmikko and T. Braysy, "Towards Cognitive Radio Systems. Main Findings from the COGNAC Project," VTT Research Notes 2575, Vuorimiehentie, 2011.

[2] J. Mitola and G. Q. M. Jr., "Cognitive Radio: Making Software Radio More Personal," *Personal Communications, IEEE,* vol. VI, no. 4, pp. 13-18, 1999.

[3] G. Ghosh, P. Das and C. Subhajit, "Simulation and Analysis of Cognitive Radio using MATLAB," *International Journal of Next-Generation Networks (IJNGN),* vol. 6, no. 2, 2014.

[4] S. Haykin, "Cognitive Radio: Brain-Empowered Wireless Communications," *IEEE Journal on Select Areas in Communications (J-SAC),* vol. 23, no. 2, pp. 201-220, 2005.

[5] V. Lal, "VHDL Coding Tips and Tricks," 27 March 2010. [Online]. Available: http://vhdlguru.blogspot.in/.

[6] B. Fette, Cognitive Radio Technology, San Francisco: Elsevier, 2006.

[7] R. Vadivelu, K. Sankaranarayan and V. Vijayakumari, "Matched Filter Based Spectrum Sensing for Cognitive Radio at Low Signal to Noise Ratio," *Journal of Theoretical and Applied Information Technology,* vol. 62, no. 1, pp. 107-113, 2014.

[8] F. M. Salem, M. H. Ibrahim, I. A. Ali and I. I. Ibrahim, "Matched Filter based spectrum sensing for Secure Cognitive Radio Network Communications," *International Journal of Computer Applications,* vol. 87, no. 18, pp. 41-46, 2014.

[9] K. Chapman, P. Hardy, A. Miller and M. George, "CDMA Matched Filter Implementation in Virtex Devices," Xilinx, 2001.

[10] J. Mitolla III and G. Q. Maguire, "Cognitive Radio: Making Software Radios More Personal," IEEE Personal Communication, 1999.

APPENDIX

%............Matched Filter Detection………..%

```
clc;
close all;
clear all;
fc=[1e5 2e5 3e5 4e5 5e5];
fs=12e5;
fm=2e3;
Ts=1/fs;
T=Ts:Ts:1/fm;
SNR_db=15;
SNR=10^(0.5*SNR_db)/0.5;
sigma=sqrt(0.5/SNR);
x=0.5*sin(2*pi*fm*T);
plot(x);
y1=0.6*(1+x).*sin(2*pi*fc(1)*T);
y2=0.2*(1+x).*sin(2*pi*fc(2)*T);
y3=0.7*(1+x).*sin(2*pi*fc(3)*T);
y4=0.4*(1+x).*sin(2*pi*fc(4)*T);
y5=0.3*(1+x).*sin(2*pi*fc(5)*T);
y=y1+y5;
yT=y + sigma*randn(1,length(T));
figure; plot(yT);
Pxx = periodogram(y);
Hpsd = dspdata.psd(Pxx,'fs',fs);
figure;
plot(Hpsd);
```

---------Energy Detection Top Module---------------

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity energy_top is
     Port ( clk, rst : in  STD_LOGIC;
            energy_out : out  STD_LOGIC_VECTOR (7 downto 0));
end energy_top;

architecture Behavioral of energy_top is

component PRSG
generic ( width: integer:= 4);
   Port ( clk : in  STD_LOGIC;
          random_num : out  STD_LOGIC_VECTOR (width-1 downto 0));
end component;

component mult
     Port ( num1, num2 : in  STD_LOGIC_VECTOR (3 downto 0);
           product : out  STD_LOGIC_VECTOR (7 downto 0));
end component;

component add
      Port ( cin : in  STD_LOGIC;
            x,y : in  STD_LOGIC_VECTOR(7 downto 0);
            r: out  STD_LOGIC_VECTOR(7 downto 0);
      cout : OUT  STD_LOGIC);
end component;
```

```
component reg
      generic ( width: integer:= 8);
      Port ( rst, clk, load : in  STD_LOGIC;
             input : in  STD_LOGIC_VECTOR (width-1 downto 0);
             output : out  STD_LOGIC_VECTOR (width-1 downto 0));
end component;

component counter
generic (n: NATURAL :=4);
      port (clk : in std_logic;
      count : buffer std_logic);
end component;

component mux
      Port ( d0, d1 : in  STD_LOGIC_VECTOR (7 downto 0);
             sel : in  STD_LOGIC;
             mux_out : out  STD_LOGIC_VECTOR (7 downto 0));
end component;

component reg1
      generic ( width: integer:= 8);
      Port ( clk : in  STD_LOGIC;
             input : in  STD_LOGIC_VECTOR (width-1 downto 0);
             output : out  STD_LOGIC_VECTOR (width-1 downto 0));
end component;

signal I0 : std_logic_vector(3 downto 0);
signal I1, I3, I2, I6, I7 : std_logic_vector(7 downto 0);
signal I5,I9: std_logic;
signal I8 : std_logic := '0';

begin

G1: PRSG PORT MAP(clk=>clk, random_num=>I0);
G2: mult PORT MAP(num1=>I0, num2=>I0, product=>I1);
G3: add PORT MAP(cin=>I8, x=>I1, y=>I3, r=>I2, cout=>I9);
G4: reg PORT MAP(rst=>I5, clk=>clk, load=>rst, input=>I2, output=>I3);
G5: counter PORT MAP(clk=>clk, count=>I5);
G7: mux PORT MAP(d0=>I7, d1=>I2, sel=>I5, mux_out=>I6);
G8: reg1 PORT MAP(clk=>clk, input=>I6, output=>I7);

energy_out <= I7;

end Behavioral;
```

--------------PRSG CODE------------

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity PRSG is
generic ( width: integer:= 4);
    Port ( clk : in  STD_LOGIC;
           random_num : out  STD_LOGIC_VECTOR (width-1 downto 0));
end PRSG;

architecture Behavioral of PRSG is
begin
process(clk)
variable rand_temp: STD_LOGIC_VECTOR(width-1 downto 0):= ("1000");
```

```
variable temp: std_logic := '0';
begin
       if(rising_edge(clk)) then
            temp := rand_temp(width-1) xor rand_temp(width-2);
            rand_temp(width-1 downto 1) := rand_temp(width-2 downto 0);
            rand_temp(0) := temp;
       end if;

      random_num <= rand_temp;

end process;
end Behavioral;
```

----------CODE FOR MULTIPLIER------------

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;


entity mult is
    port(num1, num2: in std_logic_vector(3 downto 0);
         product: out std_logic_vector(7 downto 0));
end mult;

architecture Behavioral of mult is
begin
     product <= std_logic_vector(unsigned(num1) * unsigned(num2));
end architecture Behavioral;
```

----------CODE FOR 8-BIT ADDER----------

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity add is
     Port ( cin : in  STD_LOGIC;
            x,y : in  STD_LOGIC_VECTOR(7 downto 0);
            r: out  STD_LOGIC_VECTOR(7 downto 0);
            cout : OUT  STD_LOGIC);
end add;

architecture Behavioral of add is
signal c1,c2,c3,c4,c5,c6,c7 : std_logic;
component fulladd
     Port ( cin,x,y : in  STD_LOGIC;
            s, cout : out  STD_LOGIC);
end component;

begin
stage0: fulladd PORT MAP(cin,x(0),y(0),r(0),c1);
stage1: fulladd PORT MAP(c1,x(1),y(1),r(1),c2);
stage2: fulladd PORT MAP(c2,x(2),y(2),r(2),c3);
stage3: fulladd PORT MAP(c3,x(3),y(3),r(3),c4);
stage4: fulladd PORT MAP(c4,x(4),y(4),r(4),c5);
stage5: fulladd PORT MAP(c5,x(5),y(5),r(5),c6);
stage6: fulladd PORT MAP(c6,x(6),y(6),r(6),c7);


stage7: fulladd PORT MAP(c7,x(7),y(7),r(7),cout);
end Behavioral;
```

-----------CODE FOR 8-BIT REGISTER--------

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
entity reg is
    generic ( width: integer:= 8);
    Port ( rst, clk, load : in  STD_LOGIC;
    input : in  STD_LOGIC_VECTOR (width-1 downto 0);
    output : out  STD_LOGIC_VECTOR (width-1 downto 0));
end reg;

architecture Behavioral of reg is
begin
      process(rst,clk,load)
      begin
            if(load='1') then
                output<= "00000000";
            elsif(clk'event and clk='1') then
               if(rst ='0') then
                     output<= input;
               end if;
            end if;

      end process;

end Behavioral;
```

--------------CODE FOR COUNTER----------

```
library IEEE;
use IEEE.std_logic_1164.all, IEEE.numeric_std.all;

entity counter is
generic (n: NATURAL :=4);
         port (clk : in std_logic;
         count : buffer std_logic);
end counter;



architecture behavioural of counter is

begin
     process (clk) IS
     variable cnt : unsigned (n-1 downto 0) := "0000";
begin
     if rising_edge(clk) then
           cnt := cnt + 1;
     end if;

     if cnt = "1111" then
         count <= '1';
     else


          count <= '0';
       end if;
 end process;
 end architecture behavioural;
```

------CODE FOR MULTIPLEXER------

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity mux is
      Port ( d0, d1 : in  STD_LOGIC_VECTOR (7 downto 0);
             sel : in  STD_LOGIC;
             mux_out : out  STD_LOGIC_VECTOR (7 downto 0));
end mux;

architecture Behavioral of mux is
begin
process(d0,d1,sel)
begin
     if (sel = '0') then
         mux_out <= d0;
     else mux_out <= d1;
end if;

end process;
end Behavioral;
```